Towards Efficient Axiom Pinpointing of $\mathcal{EL}^+$ Ontologies

M. Fareed Arif[1] and Joao Marques-Silva[1,2]

[1] CASL, University College Dublin, Ireland
muhammad.arif.1@ucdconnect.ie, jpms@ucd.ie
[2] INESC-ID, IST, ULisboa, Portugal

Abstract. The $\mathcal{EL}$ family of Description Logics (DLs) has been the subject of interest in recent years. On the one hand, these DLs are tractable, but fairly inexpressive. On the other hand, these DLs can be used for designing different classes of ontologies, most notably ontologies from the medical domain. Unfortunately, building ontologies is error-prone. As a result, inferable subsumption relations among concepts may be unintended. In recent years, the problem of axiom pinpointing has been studied with the purpose of providing minimal sets of axioms that explain unintended subsumption relations. For the concrete case of $\mathcal{EL}$ and $\mathcal{EL}^+$, the most efficient approaches consist of encoding the problem into propositional logic, specifically as a Horn formula, which is then analyzed with a dedicated algorithm. This paper builds on this earlier work, but exploits the important relationship between minimal axioms sets and minimal unsatisfiable subformulas in the propositional domain. In turn, this relationship allows applying a vast body of recent work in the propositional domain to the concrete case of axiom pinpointing for $\mathcal{EL}$ and its variants. From a practical perspective, the algorithms described in this paper are often several orders of magnitude more efficient that the current state of the art in axiom pinpointing for the $\mathcal{EL}$ family of DLs.

1 Introduction

Axiom pinpointing denotes the problem of computing one minimal axiom set (denoted *MinA*), which explains a subsumption relation in an ontology [29]. Axiom pinpointing for different description logics (DLs) has been studied extensively over the last decade [29,25,20,5,7,15,34,30,9,31,6,21,24,32], but some of the algorithms used can be traced back to the mid 90s [4]. Description of logics of interest have included $\mathcal{ALC}$, $\mathcal{SHIF}$, $\mathcal{SHOIN}$, in addition to the $\mathcal{EL}$ family of lightweight DLs. More recent work has focused on the tractable, albeit fairly inexpressive, $\mathcal{EL}$ family of description logics (DLs). The reason for this interest is that the $\mathcal{EL}$ family of DLs finds important applications, that include designing of medical ontologies. Besides computing one MinA, axiom pinpointing is also concerned with computing all MinAs [31,32] or computing MinAs on demand [7][1].

Original work on axiom pinpointing for the $\mathcal{EL}$ family of DLs used the well-known labeling-based classification algorithm [8,7] to find all MinAs for the $\mathcal{EL}$ family of DLs.

[1] Following existing nomenclature, MinA denotes a single axiom set, whereas MinAs denotes multiple axiom sets [7,31].

The proposed approach [8,7] generates a worst-case exponential propositional size formula, which is then used for computing all MinAs by finding all the minimal models of this formula. More recent work [31,32] proposed an encoding of the problem of computing all MinAs to a propositional Horn formulae. As an important additional result, it was shown that the resulting formulas are exponentially more compact than earlier work in the worst case. In addition, this work proposed dedicated algorithms for computing MinAs, based on propositional satisfiability (SAT) solving, but exploiting techniques used in AllSMT algorithms [16]. Although effective at computing MinAs, these dedicated algorithms often fail to enumerate all MinAs to completion, or proving that no additional MinAs exist. Nevertheless, the practical application of $\mathcal{EL}^+$ in medical ontologies and the need for axiom pinpointing motivate more efficient approaches to be developed.

The main contribution of our work is to show that the computation of MinAs can be related with the extraction of minimal unsatisfiable subformulas (MUS) of the Horn formula encoding proposed in earlier work [31,32]. More concretely, a subformula of this encoding is an MUS if and only if it represents one MinA [31,32]. Although this connection is straightforward in hindsight, we point out that it had not been investigated since the work was first published in 2009 [31]. The relationship between MUSes and MinAs allows tapping on the large recent body of work on extracting MUSes, but also on Minimal Correction Subsets (MCSes), as well as their minimal hitting set relationship [28,13,10,18,11,17,26,19], which for the propositional case allows exploiting the performance of modern SAT solvers. The relationship also allows exploring the vast body of recent work on solving maximum satisfiability (MaxSAT) [1,22] and on enumerating MaxSAT solutions [23]. The main practical consequences of this insight is that by exploiting the Horn formulae encoding proposed in earlier work [31,32] we are able to compute the set of MinAs for the vast majority of existing problem instances, and most often with many orders of magnitude performance improvements over what is currently the state of the art [31,32].

The paper is organized as follows. Section 2 overviews essential definitions and introduces the notation used throughout the paper. Section 3 overviews existing work on axiom pinpointing for the $\mathcal{EL}$ family of DLs. Section 4 briefly overviews recent work on MUS enumeration, detailing the approach used in the paper. Section 5 relates SAT-based axiom pinpointing in $\mathcal{EL}^+$ with MUS extraction and enumeration for propositional formulae, and summarizes the organization of a new $\mathcal{EL}^+$ axiom pinpointing tool, EL2MCS. Experimental results on well-known problem instances are analyzed in Section 6. Finally, the paper concludes in Section 7.

2 Preliminaries

This section introduces the notation and background material used throughout the paper, both related with description logics, namely the $\mathcal{EL}$ family of DLs, and with propositional satisfiability. Section 2.1 briefly overviews $\mathcal{EL}^+$. Afterwards, Section 2.2 summarizes work on axiom pinpointing in $\mathcal{EL}^+$. Finally, Section 2.3 reviews SAT-related definitions.

	Syntax	Semantics
top	$\top$	$\Delta^{\mathcal{I}}$
conjunction	$X \sqcap Y$	$X^{\mathcal{I}} \cap Y^{\mathcal{I}}$
existential restriction	$\exists r.X$	$\{x \in \Delta^{\mathcal{I}} \mid y \in \Delta^{\mathcal{I}} : (x, y) \in r^{\mathcal{I}} \wedge y \in X^{\mathcal{I}}\}$
general concept inclusion	$X \sqsubseteq Y$	$X^{\mathcal{I}} \subseteq Y^{\mathcal{I}}$
role inclusion	$r_1 \circ \cdots \circ r_n \sqsubseteq s$	$r_1^{\mathcal{I}} \circ \cdots \circ r_n^{\mathcal{I}} \subseteq s^{\mathcal{I}}$

Table 1: Syntax and semantics of $\mathcal{EL}^+$

2.1 Lightweight Description Logic $\mathcal{EL}^+$

This section follows standard descriptions of $\mathcal{EL}^+$ [7,31]. $\mathcal{EL}^+$ belongs to the $\mathcal{EL}$ family of lightweight description logics. Starting from a set $\mathsf{N_C}$ of *concept names* and a set $\mathsf{N_R}$ of *role names*, *concept descriptions* in $\mathcal{EL}^+$ are defined inductively using the *constructors* shown in the top part of Table 1. (As standard in earlier work, uppercase letters X, X_i, Y, Y_i denote generic concepts, uppercase letters C, C_i, D, D_i, E, E_i denote concept names and lowercase letters r, r_i, s denote role names.) A *TBox* (or *ontology*) in $\mathcal{EL}^+$ is a finite set of *general concept inclusion* (GCI) and *role inclusion* (RI) axioms. The syntax of GCIs and RIs is shown in the bottom part of Table 1. For a TBox $\mathcal{T}$, $\mathsf{PC}_{\mathcal{T}}$ denotes the set of *primitive concepts* of $\mathcal{T}$, representing the smallest set of concepts that contain: (i) the top concept $\top$; and (ii) all the concept names used in $\mathcal{T}$. $\mathsf{PR}_{\mathcal{T}}$ denotes the set of *primitive roles* of $\mathcal{T}$, representing the set of all role names used in $\mathcal{T}$. In addition, and for convenience, $X \equiv Y$ corresponds to the two GCIs $X \sqsubseteq Y$ and $Y \sqsubseteq X$. Finally, and throughout the paper, $\mathcal{A}$ denotes a set of assertions.

The semantics of $\mathcal{EL}^+$ is defined in terms of *interpretations*. An interpretation $\mathcal{I}$ is a tuple $(\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, where $\Delta^{\mathcal{I}}$ represents the domain, i.e. a non-empty set of individuals, and the interpretation function $\cdot^{\mathcal{I}}$ maps each concept name $C \in \mathsf{N_C}$ to a set $C^{\mathcal{I}} \in \Delta^{\mathcal{I}}$, and maps each role name $r \in \mathsf{N_R}$ to a binary relation $r^{\mathcal{I}}$ defined on $\Delta^{\mathcal{I}}$, i.e. $r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$. The third column of Table 1 details the inductive definitions of $\cdot^{\mathcal{I}}$ for arbitrary concept descriptions. An interpretation $\mathcal{I}$ is a *model* of a TBox $\mathcal{T}$ if and only if the conditions in semantics (third) column of Table 1 are respected for every GCI and RI axiom in $\mathcal{T}$.

The main inference problem for $\mathcal{EL}^+$ is the subsumption problem:

Definition 1 (Concept Subsumption). *Let C, D represent two $\mathcal{EL}^+$ concept descriptions and let $\mathcal{T}$ represent an $\mathcal{EL}^+$ TBox. C is subsumed by D w.r.t. $\mathcal{T}$ (denoted $C \sqsubseteq_{\mathcal{T}} D$) if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ in every model $\mathcal{I}$ of $\mathcal{T}$.*

Subsumption algorithms assume that a given TBox $\mathcal{T}$ is normalized [3]. Normalization can be viewed as the process of breaking complex GCIs into simpler ones. It is well-known that a TBox $\mathcal{T}$ can be normalized in linear time [3]. Moreover, given a normalized TBox, concept subsumption can be determined in polynomial time [3].

Theorem 1 (Theorem 1 in [7]). *Given a TBox $\mathcal{T}$ in normal form, the subsumption algorithm runs in polynomial time on the size of $\mathcal{T}$.*

$$
\mathcal{T}_{\mathrm{MG}} := \begin{array}{l}
\{\mathrm{Endocarditis} \sqsubseteq \mathrm{Inflammation} \sqcap \exists \mathrm{hasLoc.Endocardium}, \\
\mathrm{Inflammation} \sqsubseteq \mathrm{Disease} \sqcap \exists \mathrm{actsOn.Tissue}, \\
\mathrm{Endocardium} \sqsubseteq \mathrm{Tissue} \sqcap \exists \mathrm{contIn.HeartValve}, \\
\mathrm{HeartValve} \sqsubseteq \exists \mathrm{contIn.Heart}, \\
\mathrm{HeartDisease} \equiv \mathrm{Disease} \sqcap \exists \mathrm{hasLoc.Heart}, \\
\mathrm{contIn} \circ \mathrm{contIn} \sqsubseteq \mathrm{contIn}, \\
\mathrm{hasLoc} \circ \mathrm{contIn} \sqsubseteq \mathrm{hasLoc}\}
\end{array}
$$

$$
\mathcal{A}_{\mathrm{MG}} := \begin{array}{l}
\{\mathrm{Endocarditis} \sqsubseteq \mathrm{Inflammation}, \\
\mathrm{Inflammation} \sqsubseteq \mathrm{Disease}, \\
\mathrm{Endocardium} \sqsubseteq \mathrm{Tissue}, \\
\mathrm{HeartDisease} \sqsubseteq \mathrm{Disease}, \\
\mathrm{Endocarditis} \sqsubseteq \exists \mathrm{hasLoc.Endocardium}, \\
\mathrm{Inflammation} \sqsubseteq \exists \mathrm{actsOn.Tissue}, \\
\mathrm{Endocardium} \sqsubseteq \exists \mathrm{contIn.HeartValve}, \\
\mathrm{HeartValve} \sqsubseteq \exists \mathrm{contIn.Heart}, \\
\mathrm{HeartDisease} \sqsubseteq \exists \mathrm{hasLoc.Heart}, \\
\mathrm{Disease} \sqcap \mathrm{N} \sqsubseteq \mathrm{HeartDisease}, \\
\exists \mathrm{hasLocHeart} \sqsubseteq \mathrm{N}, \\
\mathrm{Endocarditis} \sqsubseteq \mathrm{Disease}, \\
\mathrm{Endocarditis} \sqsubseteq \exists \mathrm{actsOn.Tissue}, \\
\mathrm{Endocarditis} \sqsubseteq \exists \mathrm{hasLoc.HeartValve}, \\
\mathrm{Endocardium} \sqsubseteq \exists \mathrm{contIn.Heart}, \\
\mathrm{HeartDisease} \sqsubseteq \mathrm{N}, \\
\mathrm{Endocarditis} \sqsubseteq \exists \mathrm{hasLocHeart} \leftarrow \mathrm{N}, \\
\mathrm{Endocarditis} \sqsubseteq \mathrm{N}, \\
\mathbf{Endocarditis} \sqsubseteq \mathbf{HeartDisease}\}
\end{array}
$$

Table 2: An example $\mathcal{T}_{\mathrm{MG}}$ medical ontology and its classification.

A classification of TBox $\mathcal{T}$ represents all subsumption relations between concept names in $\mathcal{T}$. Similarly to subsumption, and because of the worst-case quadratic number of subsumption relations, classification can be determined in polynomial time on the size of $\mathcal{T}$ [3,7].

Example 1 (GALEN-based Medical Ontology Example). Let us consider an $\mathcal{EL}^+$ medical ontology adapted from the GALEN medical ontology [27], and shown in Table 2. The ontology expresses a medical condition in which endocarditis is classified as a heart disease (i.e., Endocarditis $\sqsubseteq$ HeartDisease). This disease occurs due to a bacteria that damages endocardium, a tissue, that provides a protection to the heart valves. The table shows both the ontoloy and its classification. The normalized ontology consists of 11 GCI's and 2 role inclusion axioms (i.e., contIn $\circ$ contIn $\sqsubseteq$ contIn and hasLoc $\circ$ contIn $\sqsubseteq$ hasLoc). Only assertions are added to the assertion axiom set $\mathcal{A}_{\mathrm{MG}}$. Regarding the classification of this ontology, N denotes the reference to the assertion axiom Endocarditis $\sqsubseteq$ $\exists$hasLocHeart, and serves to simplify the notation of assertion axioms set.

2.2 Axiom Pinpointing in $\mathcal{EL}^+$

Axiom pinpointing is the problem of explaining unintended subsumption relations in description logics. As is the case in so many other areas of research, the goal of axiom pinpointing is to find minimal explanations for unintended subsumption relations. The importance of this research topic is illustrated by the large body of work, targeting different description logics [29,25,20,5,7,15,34,30,9,31,6,21,32]. For the concrete case of the $\mathcal{EL}$ family of DLs, two main approaches have been proposed [5,7,31,32].

Earlier work [5,7] consists in creating a pinpointing formula ϕ, and then enumerating the minimal models of ϕ. This is the approach implemented in the CEL tool [5]. The main drawback of this work is that the pinpointing formula ϕ is worst-case exponential on the size of $\mathcal{T}$, and enumeration of minimal models is NP-hard. In contrast, more recent work showed that concept subsumption can be encoded to a Horn formula, and that the axiom pinpointing problem can be solved on this Horn formula [31,32]. This approach is detailed in Section 3.

Throughout this paper, the following standard definition of MinA (and of nMinA) is assumed.

Definition 2 (nMinA/MinA). *Let $\mathcal{T}$ be an $\mathcal{EL}^+$ TBox, and let $C, D \in \mathsf{PC}_{\mathcal{T}}$ be primitive concept names, with $C \sqsubseteq_{\mathcal{T}} D$. Let $\mathcal{S}$ be a subset of $\mathcal{T}$ be such that $C \sqsubseteq_{\mathcal{S}} D$. If $\mathcal{S}$ is such that $C \sqsubseteq_{\mathcal{S}} D$ and $C \not\sqsubseteq_{\mathcal{S}'} D$ for $\mathcal{S}' \subset \mathcal{S}$, then $\mathcal{S}$ is a* minimal axiom set *(MinA) w.r.t. $C \sqsubseteq_{\mathcal{T}} D$. Otherwise, $\mathcal{S}$ is a non-minimal axiom set (nMinA) w.r.t. $C \sqsubseteq_{\mathcal{T}} D$.*

2.3 Propositional Satisfiability

Standard propositional satisfiability (SAT) definitions are assumed [12]. This includes standard definitions for variables, literals, clauses and CNF formulas. Formulas are represented by $\mathcal{F}$, $\mathcal{M}$, $\mathcal{M}'$, $\mathcal{C}$ and $\mathcal{C}'$, but also by ϕ and ψ. Horn formulae are such that every clause contains at most one positive literal. It is well-known that SAT on Horn formulae can be decided in linear time [14]. The paper explores both MUSes and MCSes of propositional formulae.

Definition 3 (MUS). *$\mathcal{M} \subseteq \mathcal{F}$ is a* Minimal Unsatisfiable Subformula *(MUS) of $\mathcal{F}$ iff $\mathcal{M}$ is unsatisfiable and $\forall_{\mathcal{M}' \subsetneq \mathcal{M}} \mathcal{M}'$ is satisfiable.*

Definition 4 (MUS). *$\mathcal{C} \subseteq \mathcal{F}$ is a* Minimal Correction Subformula *(MCS) of $\mathcal{F}$ iff $\mathcal{F} \setminus \mathcal{C}$ is satisfiable and $\forall_{\mathcal{C}' \subsetneq \mathcal{C}} \mathcal{F} \setminus \mathcal{C}'$ is unsatisfiable.*

A well-known result, which will be used in the paper is the minimal hitting set relationship between MUSes and MCSes of an unsatisfiable formula $\mathcal{F}$ [28,13,10,18].

Theorem 2. *Let $\mathcal{F}$ be unsatisfiable. Then,*

1. *Each MCS of $\mathcal{F}$ is a minimal hitting set of the MUSes of $\mathcal{F}$.*
2. *Each MUS of $\mathcal{F}$ is a minimal hitting set of the MCSes of $\mathcal{F}$.*

As highlighted in Section 4, Theorem 2 forms the basis of *all* existing approaches for enumerating MUSes [10,18,17,26]. Moreover, the importance of set duality relationships in combinatorial optimization is highlighted by recent work [35].

3 SAT-Based Axiom Pinpointing in $\mathcal{EL}^+$

This section reviews EL2SAT, a recently proposed approach for axiom pinpointing of $\mathcal{EL}^+$ ontologies using a Horn formula encoding [31,32]. The EL2SAT approach can be divided into two main components. First, the axiom pinpointing problem is encoded as a Horn formula which is polynomial on the size of $\mathcal{T}$. Second, the MinAs are computed using a dedicated algorithm, exploiting ideas from early work on AllSMT [16].

The EL2SAT axiom pinpointing approach for $\mathcal{EL}^+$ can be summarized as follows [31,32]:

A. Encode the classification of TBox $\mathcal{T}$ by running the standard classification algorithm and adding Horn clauses for representing every non-trivial axiom or assertion as a set of Horn clauses. The resulting Horn formula is denoted $\phi_{\mathcal{T}}$.
B. Encode the complete classification DAG of the input normalized ontology $\mathcal{T}$ as a Horn formula $\phi_{\mathcal{T}}^{all}$.
C. Associate a Boolean variable $s_{[a_i]}$ with each assertion a_i in the classification of $\mathcal{T}$ and create a clause $s_{[a_i]} \rightarrow \mathcal{EL}^+2\mathsf{SAT}(a_i)$, where $\mathcal{EL}^+2\mathsf{SAT}(a_i)$ is the set of clauses associated with a_i in $\phi_{\mathcal{T}}^{all}$. The resulting formula is denoted $\phi_{\mathcal{T}(so)}$.
D. Encode the *pinpointing-only* problem as another Horn formula $\phi_{\mathcal{T}(\mathrm{po})}^{\mathrm{all}}$. For the purposes of this paper, this is the formula that is of interest and so the one analyzed in greater detail.

The construction of the target Horn formula $\phi_{\mathcal{T}(\mathrm{po})}^{\mathrm{all}}$ mimics the construction of the other formulas, in that the classification procedure is executed. The formula is constructed as follows:

1. For every RI axiom, create an axiom selector variable $s_{[a_i]}$. For trivial GCI of the form $C \sqsubseteq C$ or $C \sqsubseteq \top$, $s_{[a_i]}$ is constant $\mathtt{true}$. For each non-trivial GCIs, add an axiom selector variable $s_{[a_i]}$.
2. During the execution of the classification algorithm, for every application of a rule (concretely r) generating some assertion $\mathsf{gen}(r)$ (concretely a_i), add to $\phi_{\mathcal{T}(\mathrm{po})}^{\mathrm{all}}$ a clause of the form,

$$\left(\bigwedge_{a_j \in \mathsf{ant}(r)} s_{[a_j]} \right) \rightarrow s_{[a_i]} \qquad (1)$$

where $s_{[a_i]}$ is the selector variable for a_i and $\mathsf{ant}(r)$ are the antecedents of a_i with respect to rule r.

As explained in earlier work, the encoding procedure ensures that each rule application is applied only once. Finally, for the concrete case of axiom pinpointing, specify the assumption list $\{\neg s_{[C_i \sqsubseteq D_i]}\} \cup \{s_{[a_i]} \mid a_i \in \mathcal{T}\}$. This assumption list is manipulated by the dedicated algorithm proposed in earlier work [31,32].

The following theorem is fundamental for earlier work [31,32], and is extended in the next section to related MinAs with MUSes of propositional formulae.

Theorem 3 (Theorem 3 in [32]). *Given an $\mathcal{EL}^+$ TBox $\mathcal{T}$, for every $\mathcal{S} \subseteq \mathcal{T}$ and for every pair of concept names $C, D \in \mathsf{PC}_{\mathcal{T}}$, $C \sqsubseteq_{\mathcal{S}} D$ if and only if the Horn propositional*

formula,

$$\phi^{\text{all}}_{\mathcal{T}(\text{po})} \wedge_{ax_i \in \mathcal{S}} (s_{[ax_i]}) \wedge (\neg s_{[C \sqsubseteq D]}) \tag{2}$$

is unsatisfiable.

Example 2 (GALEN Medical Ontology SAT Encoding). For the ontology in Example 1, Table 3 illustrates the construction of the formulas described earlier in this section. Regarding the $s_{[a_i]}$ variables, the original set is $\{s_1, \ldots, s_{13}\}$. These represent the axioms in the normalized ontology. The remaining ones result from the classification procedure. It it important to note that we do not expand $\mathcal{EL}^+2\mathsf{SAT}(a_i)$ in $\phi_{\mathcal{T}_{\text{MG}(so)}}$. Moreover, a selection variable is assigned to each role inclusion axiom.

4 Enumeration of MUSes

The problem of enumerating all the MUSes of an unsatisfiable formula has been studied in different settings [28,13,10,18,17,26], starting with the seminal work of Reiter [28].

Although the enumeration of MCSes can be achieved in a number of different ways [19], the enumeration of MUSes is believed to be far more challenging. Intuitively, the main difficulty is how to block one MUS and then use a SAT solver (or some other decision procedure) to compute the next MUS. This difficulty with MUS enumeration motivated a large body of work exploiting a fundamental relationship between MUSes and MCSes. Indeed, it is well-known (e.g. see Theorem 2) that MCSes are minimal hitting sets of MUSes, and MUSes are minimal hitting sets of MCSes [28,13,10,18]. As a result, early approaches [10,18] for enumeration of MUSes were organized as follows: (i) enumerate all MCSes of a formula; (ii) compute the minimal hitting sets of the MCSes.

One problem with early approaches is that all MCSes need to be enumerated *before* the first MUS is computed. As a result, more recent work proposed alternative approaches which enable both MUSes and MCSes to be computed while enumeration of MUSes (and MCSes) takes place [17,26]. Nevertheless, if the number of MCSes is manageable, earlier work is expected to be more efficient [18]. It should noted that, all existing approaches for MUS enumeration relate with Theorem 2, in that enumeration of MUSes is achieved by explicitly or implicitly computing the minimal hitting sets of all MCSes.

Regarding MCS enumeration, two main approaches have been studied. One exploits MaxSAT enumeration [18,23]. More recent work proposes dedicated MCS extraction algorithms, also capable of enumerating MCSes [19]. The approach proposed in this paper exploits MaxSAT enumeration.

5 Axiom Pinpointing with MUS Extraction

This section shows that the computation of MinAs can be related with MUS enumeration of the Horn formula $\phi^{all}_{\mathcal{T}(po)}$. It then briefly overviews existing approaches for MUS enumeration, and concludes by summarizing the MUS enumeration approach used in this paper.

$$
\phi_{\mathcal{T}_{\text{MG}(so)}} := \begin{array}{l}
s_1 \to \text{Endocarditis} \sqsubseteq \text{Inflammation} \\
s_2 \to \text{Inflammation} \sqsubseteq \text{Disease} \\
s_3 \to \text{Endocardium} \sqsubseteq \text{Tissue} \\
s_4 \to \text{HeartDisease} \sqsubseteq \text{Disease} \\
s_5 \to \text{Endocarditis} \sqsubseteq \exists \text{hasLoc.Endocardium} \\
s_6 \to \text{Inflammation} \sqsubseteq \exists \text{actsOn.Tissue} \\
s_7 \to \text{Endocardium} \sqsubseteq \exists \text{contIn.HeartValve} \\
s_8 \to \text{HeartValve} \sqsubseteq \exists \text{contIn.Heart} \\
s_9 \to \text{HeartDisease} \sqsubseteq \exists \text{hasLoc.Heart} \\
s_{10} \to \text{Disease} \sqcap \text{N} \sqsubseteq \text{HeartDisease} \\
s_{11} \to \exists \text{hasLocHeart} \sqsubseteq \text{N} \\
s_{12} \to \text{contIn} \circ \text{contIn} \sqsubseteq \text{contIn} \\
s_{13} \to \text{hasLoc} \circ \text{contIn} \sqsubseteq \text{hasLoc} \\
s_{14} \to \text{Endocarditis} \sqsubseteq \text{Disease} \\
s_{15} \to \text{Endocarditis} \sqsubseteq \exists \text{actsOn.Tissue} \\
s_{16} \to \text{Endocarditis} \sqsubseteq \exists \text{hasLoc.HeartValve} \\
s_{17} \to \text{Endocardium} \sqsubseteq \exists \text{contIn.Heart} \\
s_{18} \to \text{HeartDisease} \sqsubseteq \text{N} \\
s_{19} \to \text{Endocarditis} \sqsubseteq \exists \text{hasLocHeart} \leftarrow \text{N} \\
s_{20} \to \text{Endocarditis} \sqsubseteq \text{N} \\
s_{21} \to \textbf{Endocarditis} \sqsubseteq \textbf{HeartDisease}
\end{array}
$$

$$
\phi^{all}_{\mathcal{T}_{\text{MG}(po)}} := \begin{array}{l}
\{s_1 \wedge s_2 \to s_4, \\
s_6 \wedge s_1 \to s_{15}, \\
s_{13} \wedge s_7 \wedge s_5 \to s_{16}, \\
s_{12} \wedge s_8 \wedge s_7 \to s_{17}, \\
s_{11} \wedge s_9 \to s_{18}, \\
s_{13} \wedge s_{16} \wedge s_8 \to s_{19}, \\
s_{13} \wedge s_{17} \wedge s_5 \to s_{19}, \\
s_{11} \wedge s_{19} \to s_{20}, \\
s_{10} \wedge s_{14} \wedge s_{20} \to s_{21}, \\
s_4 \wedge s_{21} \to s_{14}, \\
s_9 \wedge s_{21} \to s_{19}\}
\end{array}
$$

$$
\phi^{all}_{\mathcal{T}_{\text{MG}}} := \phi^{all}_{\mathcal{T}_{\text{MG}(po)}} \wedge \bigwedge_{1 \leq i \leq 13} (s_i) \wedge (\neg s_{21})
$$

$$
\text{MinA} := \{s_1, s_5, s_8, s_{10}, s_{11}, s_{13}\}
$$

Table 3: The $\mathcal{T}_{\text{MG}}$ Horn encoding [31,32].

5.1 MinAs as MUSes

Although not explicitly stated, the relation between axiom pinpointing and MUS extraction has been apparent in earlier work [7,31,32]. Indeed, from the encoding of axiom pinpointing to Horn formula, it is immediate the following result, that relates Theorem 3 (Theorem 3 in [32]) with the dedicated algorithm proposed in earlier work [31,32]:

Theorem 4. *Given an $\mathcal{EL}^+$ TBox $\mathcal{T}$, for every $\mathcal{S} \subseteq \mathcal{T}$ and for every pair of concept names $C, D \in \mathsf{PC}_{\mathcal{T}}$, $\mathcal{S}$ is a MinA of $C \sqsubseteq_{\mathcal{S}} D$ if and only if the Horn propositional formula,*

$$\phi^{\text{all}}_{\mathcal{T}(\text{po})} \wedge_{ax_i \in \mathcal{S}} (s_{[ax_i]}) \wedge (\neg s_{[C \sqsubseteq D]}) \tag{3}$$

is minimally unsatisfiable.

Proof. [Sketch] By Theorem 3, $C \sqsubseteq_{\mathcal{S}} D$ if and only if the associated Horn formula (3) is unsatisfiable. For a MinA $\mathcal{S} \subseteq \mathcal{T}$, minimal unsatisfiability of (3) (with $\mathcal{T}$ replaced by $\mathcal{S}$) results from the MinA computation algorithm proposed in earlier work [31,32]. □

Based on Theorem 4 and the MUS enumeration approaches summarized in Section 4, we can now outline our approach based on MUS enumeration.

Earlier work [31,32] explicitly enumerates assignments to the $s_{[ax_i]}$ variables in a AllSMT-inspired approach [16]. In contrast, our approach is to model the problem has partial maximum satisfiability (MaxSAT), and enumerate the MUSes of the MaxSAT problem formulation.

All clauses in $\phi^{\text{all}}_{\mathcal{T}(\text{po})}$ are declared as hard clauses, i.e. they must be satisfied. Observe that, by construction, $\phi^{\text{all}}_{\mathcal{T}(\text{po})}$ is satisfiable. In addition, the constraint $C \sqsubseteq_{\mathcal{T}} D$ is encoded with another hard clause, namely $(\neg s_{[C \sqsubseteq_{\mathcal{T}} D]})$. Finally, the variable associated with each axiom ax_i, $s_{[ax_i]}$ denotes a *unit soft clause*. The intuitive justification is that the goal is to include as many axioms as possible, leaving out a minimal set which if included will cause the complete formula to be unsatisfiable. Thus, each of these sets represents an MCS of the MaxSAT problem formulation, but also a minimal set of axioms that needs to be dropped for the subsumption relation not to hold. MCS enumeration can easily be implemented with a MaxSAT solver [18,23] or with a dedicated algorithm [19]. Moreover, we can now use minimal hitting set dualization [28,13,10,18] to obtain the MUSes we are looking for, starting from the previously computed MCSes. This is the approach implemented in this paper.

Example 3. For the ontology in Example 1 and Example 2, Table 4 summarizes the MaxSAT formulation, as well as an example MUS. The hard clauses are given by $\phi^{\text{all}}_{\mathcal{T}(\text{po})}$ and by $(\neg s_{[C \sqsubseteq D]}$. Each positive unit soft clauses is given by the selection variable associated with each of the original axioms. The EL2MCS approach starts by computing the MCSes, using a MaxSAT-based approach, and then computes the MUSes by minimal hitting set dualization (e.g. see Theorem 2).

Related Work. Reiter's work on computing minimal hitting sets [28] is used to find all the justifications in earlier work [15,37], which exploits hitting set trees (HST). This approach starts by computing a single justification by using either a standard

$\phi_{\mathcal{H}} := \{\phi^{all}_{\mathcal{T}_{\mathrm{MG}}(po)}\} \cup \{\neg s_{21}\}$
$\phi_{\mathcal{S}} := \{s_1, s_2, \ldots, s_{13}\}$
$\Omega := \langle \phi_{\mathcal{H}}, \phi_{\mathcal{S}} \rangle$
MUS $:= \{s_1, s_5, s_8, s_{10}, s_{11}, s_{13}\}$

Table 4: The MaxSAT formulation.

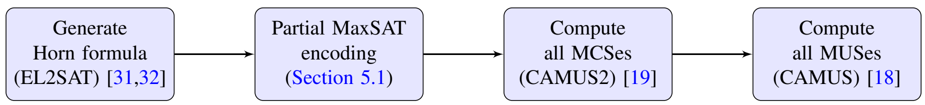

Fig. 1: The EL2MCS tool

blackbox method, either a sliding window deletion based-deletion algorithm (SINGLE_JUST_ALG in [15]) or a binary search based algorithm (log-extract-mina in [9]). After finding the first justification using any of these methods, the algorithm removes axioms one by one from this justification set and constructs a Hitting Set Tree (HST) that in turn serves to find all the justifications (using either SINGLE_JUST_ALG or log-extract-mina on each branch of the HST). The main drawback of this approach is that it does not scale for large size ontologies. Therefore, this method is only used in conjunction with reachability-based modules [9] and when a TBox contains only a few relevant axioms subject to an entailment.

5.2 The EL2MCS Axiom Pinpointing Approach

This section summarizes the organization of the EL2MCS tool, that exploits MUS enumeration for axiom pinpointing. The organization of EL2MCS is shown in Figure 1. The first step is similar to EL2SAT [31,32] in that a propositional Horn formula is generated. The next step, however, exploits the ideas in the previous section, and generates a partial MaxSAT encoding. As outlined earlier, we can now enumerate the MCSes of the partial MaxSAT formula. This is achieved with the CAMUS2 tool [19][2]. The final step is to exploit minimal hitting set dualization for computing all the MUSes given the set of MCSes [18]. This is achieved with the CAMUS tool[3]. It should be observed that, although MCS enumeration uses CAMUS2 (a modern implementation of the MCS enumerator in CAMUS [18], capable of handling partial MaxSAT formulae), alternative MCS enumeration approaches were considered [19], not being as efficient.

6 Experimental Results

The experiments were performed on an HPC cluster, with dual quad-core Intel Xeon 3GHz processors, with 32GB of physical memory. The tools EL2SAT and EL2MCS

[2] Available from http://logos.ucd.ie/web/doku.php?id=mcsls.
[3] Available from http://sun.iwu.edu/~mliffito/camus/.

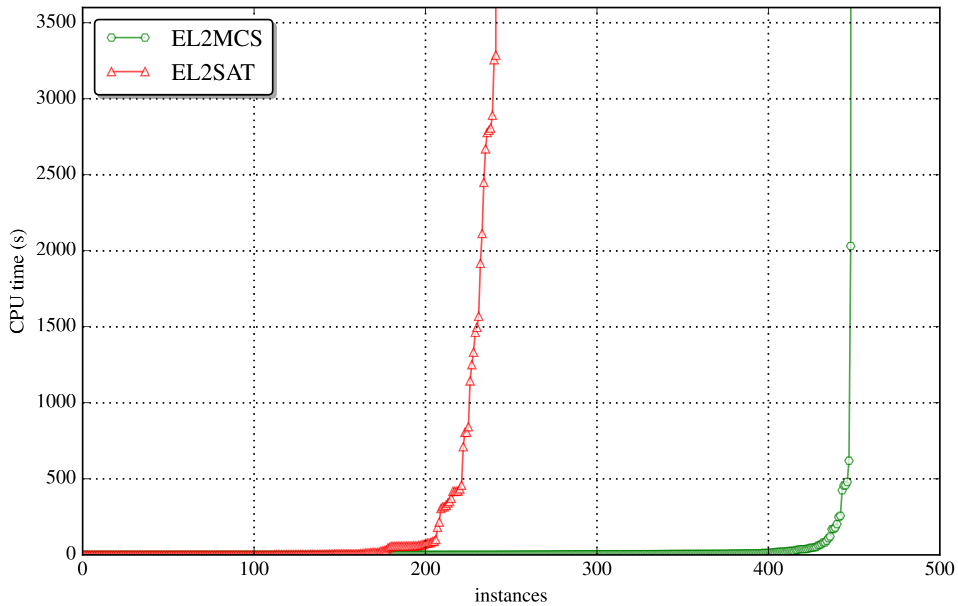

Fig. 2: Cactus plot comparing EL2MCS and EL2SAT on all problem instances, with COI reduction

were given a timeout of 3600 seconds and a memory limit of 16GB. EL2MCS uses the partial MaxSAT instances generated with the help of EL2SAT Horn propositional encoding tool[4]. The medical ontologies used in the experimentation are GALEN [27], Gene [2], NCI [33] and SNOMED-CT [36][5]. We have used the 450 subsumption query instances which are studied in earlier work [32] (and which are available from the EL2SAT website). The experimental results compare exclusively EL2SAT and EL2MCS, since EL2SAT has recently been shown to consistently outperform CEL [5], a $\mathcal{EL}^+$ axiom pinpointing tool [32]. Moreover, unless otherwise stated, the experiments consider the optimizations proposed in earlier work [31,32], and exploited in the EL2SAT tool. One example is the cone-of-influence (COI) simplification technique [32], which enables significant reductions in the obtained propositional satisfiability instances. It should be noted that cone-of-influence reduction technique can be related with $\mathcal{EL}^+$ reachability-based modularization [9].

[4] The $\mathcal{EL}^+$ to SAT encoder is denoted el2sat_all, whereas the axiom pinpointing tool is el2sat_all_mins. These tools are available from `http://disi.unitn.it/~rseba/elsat/`. Moreover, this site also contains the subsumption query instances used in the experiments.

[5] GENE, GALEN and NCI ontologies are freely available at `http://lat.inf.tu-dresden.de/~meng/toyont.html`. The SNOMED-CT ontology was requested from IHTSDO under a nondisclosure license agreement.

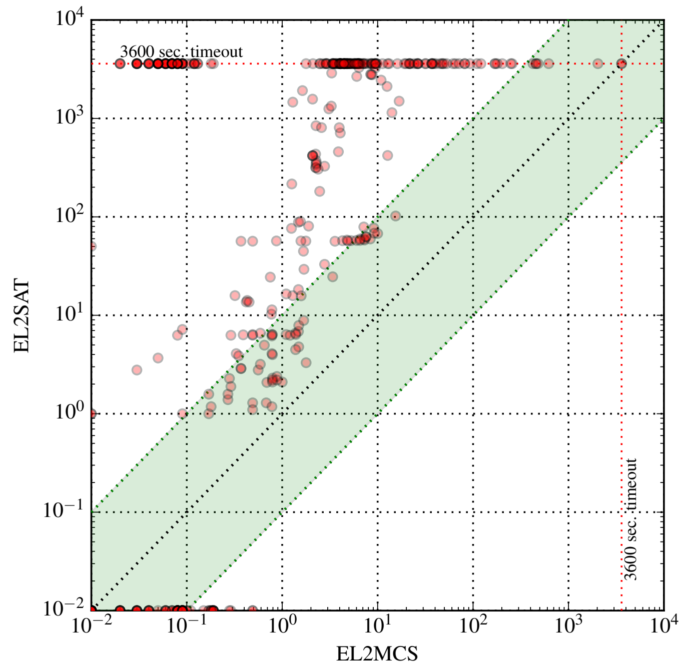

Fig. 3: Scatter plot comparing EL2MCS and EL2SAT on all problem instances, with COI reduction

Figure 2 shows a cactus plot comparing EL2SAT with EL2MCS. The performance gap between EL2SAT and EL2MCS is clear and conclusive. As summarized in Table 5, EL2SAT solves 241 out of 450 instances, whereas EL2MCS solves 448 out of 450 instances. Figure 3 shows a scatter plot comparing the two tools. As before, the performance gap between EL2SAT and EL2MCS is clear, with performance gains that often exceed one order of magnitude, and that can even exceed three orders of magnitude. More significantly, for 207 instances (i.e. $209-2$ out of 450), and in contrast with EL2MCS, EL2SAT does not terminate within the given timeout.

Table 5 summarizes the statistics of running the two tools with and without COI reduction. As can be concluded, COI reduction is far more relevant for EL2SAT than for EL2MCS. When COI reduction is used, EL2SAT outperforms EL2MCS on 16.9% of the instances. It should be noted that *all* of these instances terminate in less that 0.5 seconds for both tools, as can be concluded from Figure 3. In contrast, EL2MCS outperforms EL2SAT on 74.9% of the instances, and for 207 (i.e. $209 - 2$) of these, EL2SAT does not terminate. The statistics further support the conclusion that the extraction of MUSes provides a far more robust solution than a dedicated algorithm based on enumeration of subsets and exploiting AllSMT techniques.

Table 6 summarizes the number of times each tool computes more MUSes than the other tools, independently of being able to prove the non-existence of additional MUSes within the given timeout. As can be observed, EL2SAT is actually quite capa-

	EL2SAT	EL2MCS
# Solved	241	448
% Solved	53.6%	99.6%
# TO	209	2
% TO	46.4%	0.4%
% Wins	16.9%	74.9%

(a) With COI reduction

	EL2SAT	EL2MCS
# Solved	8	447
% Solved	1.8%	99.3%
# TO	442	3
% TO	98.2%	0.7%
% Wins	0.2%	99.3%

(b) Without COI reduction

Table 5: Statistics summarizing the results on a universe of 450 problem instances with a timeout of 3600s, with and without COI reduction.

EL2SAT	EL2MCS	Ties
2	19	429

(a) With COI reduction

EL2SAT	EL2MCS	Ties
3	71	376

(b) Without COI reduction

Table 6: Number of times a tool computes more MUSes within a timeout of 3600s, with and without COI reduction.

ble of finding most MUSes, especially when COI reduction is applied. The reason is how the algorithm is implemented and the preference to promot conflicts. However, as shown in Figure 2 and Figure 3, in many cases EL2SAT takes significantly more time to compute all MUSes, and it is often unable to prove the non-existence of additional MUSes. These results also indicate that MUSes in the axiom pinpointing instances considered are usually small. As a result, EL2SAT, which uses assumptions for the implicit set enumeration of target sets, is able to compute most MUSes in many cases. In contrast, in situations where the size of MUSes is larger, EL2SAT would be expected to be unable to reach a stage where the sets representing MUSes would be enumerated.

7 Conclusions & Future Work

Axiom pinpointing serves to identify unintended subsumption relations in DLs. For the $\mathcal{EL}$ family of DLs, there has been recent work on axiom pinpointing, the most efficient of which is based on encoding the problem to propositional Horn formulae. The main contribution of this paper is to relate axiom pinpointing with MUS extraction. As a result, this enables exploiting different MUS extraction and enumeration algorithms for the problem of axiom pinpointing. Preliminary results, obtained using off-the-shelve tools, show categorical performance gains over the current state of the art in axiom pinpointing for the $\mathcal{EL}$ family of DLs.

The results are justify further analysis of the uses of MUSes and MCSes in axiom pinpointing, for the $\mathcal{EL}$ family of DLs, but also for other DLs.

Acknowledgement. We thank the authors of EL2SAT for authorizing the use of the most recent, yet unpublished, version of their work. This work is partially supported by

SFI PI grant BEACON (09/IN.1/I2618), by FCT grant POLARIS (PTDC/EIA-CCO/-123051/2010), and by national funds through FCT with reference UID/CEC/50021/2013.

References

1. Carlos Ansótegui, Maria Luisa Bonet, and Jordi Levy. SAT-based MaxSAT algorithms. *Artif. Intell.*, 196:77–105, 2013.
2. Michael Ashburner, Catherine A. Ball, Judith A. Blake, David Botstein, Heather Butler, J. Michael Cherry, Allan P. Davis, Kara Dolinski, Selina S. Dwight, Janan T. Eppig, and et al. Gene ontology: tool for the unification of biology. *Nature genetics*, 25(1):25–29, 2000.
3. Franz Baader, Sebastian Brandt, and Carsten Lutz. Pushing the $\mathcal{EL}$ envelope. In *IJCAI*, pages 364–369, 2005.
4. Franz Baader and Bernhard Hollunder. Embedding defaults into terminological knowledge representation formalisms. *J. Autom. Reasoning*, 14(1):149–180, 1995.
5. Franz Baader, Carsten Lutz, and Boontawee Suntisrivaraporn. CEL - A polynomial-time reasoner for life science ontologies. In *IJCAR*, pages 287–291, 2006.
6. Franz Baader and Rafael Peñaloza. Axiom pinpointing in general tableaux. *J. Log. Comput.*, 20(1):5–34, 2010.
7. Franz Baader, Rafael Peñaloza, and Boontawee Suntisrivaraporn. Pinpointing in the description logic $\mathcal{EL}^+$. In *KI*, pages 52–67, 2007.
8. Franz Baader, Rafael Peñaloza, and Boontawee Suntisrivaraporn. Pinpointing in the description logic $\mathcal{EL}$. In *DL*, 2007.
9. Franz Baader and Boontawee Suntisrivaraporn. Debugging SNOMED CT using axiom pinpointing in the description logic $\mathcal{EL}^+$. In *KR-MED*, 2008.
10. James Bailey and Peter J. Stuckey. Discovery of minimal unsatisfiable subsets of constraints using hitting set dualization. In *PADL*, pages 174–186, 2005.
11. Anton Belov, Inês Lynce, and Joao Marques-Silva. Towards efficient MUS extraction. *AI Commun.*, 25(2):97–116, 2012.
12. Armin Biere, Marijn Heule, Hans van Maaren, and Toby Walsh, editors. *Handbook of Satisfiability*, volume 185 of *Frontiers in Artificial Intelligence and Applications*. IOS Press, 2009.
13. Elazar Birnbaum and Eliezer L. Lozinskii. Consistent subsets of inconsistent systems: structure and behaviour. *J. Exp. Theor. Artif. Intell.*, 15(1):25–46, 2003.
14. William F. Dowling and Jean H. Gallier. Linear-time algorithms for testing the satisfiability of propositional Horn formulae. *J. Log. Program.*, 1(3):267–284, 1984.
15. Aditya Kalyanpur, Bijan Parsia, Matthew Horridge, and Evren Sirin. Finding all justifications of OWL DL entailments. In *ISWC*, pages 267–280, 2007.
16. Shuvendu K. Lahiri, Robert Nieuwenhuis, and Albert Oliveras. SMT techniques for fast predicate abstraction. In *CAV*, pages 424–437, 2006.
17. Mark H. Liffiton and Ammar Malik. Enumerating infeasibility: Finding multiple MUSes quickly. In *CPAIOR*, pages 160–175, 2013.
18. Mark H. Liffiton and Karem A. Sakallah. Algorithms for computing minimal unsatisfiable subsets of constraints. *J. Autom. Reasoning*, 40(1):1–33, 2008.
19. Joao Marques-Silva, Federico Heras, Mikolás Janota, Alessandro Previti, and Anton Belov. On computing minimal correction subsets. In *IJCAI*, 2013.
20. Thomas Andreas Meyer, Kevin Lee, Richard Booth, and Jeff Z. Pan. Finding maximally satisfiable terminologies for the description logic $\mathcal{ALC}$. In *AAAI*, pages 269–274, 2006.
21. Kodylan Moodley, Thomas Meyer, and Ivan José Varzinczak. Root justifications for ontology repair. In *RR*, pages 275–280, 2011.

22. António Morgado, Federico Heras, Mark H. Liffiton, Jordi Planes, and Joao Marques-Silva. Iterative and core-guided MaxSAT solving: A survey and assessment. *Constraints*, 18(4):478–534, 2013.
23. António Morgado, Mark H. Liffiton, and Joao Marques-Silva. MaxSAT-based MCS enumeration. In *HVC*, pages 86–101, 2012.
24. Hai H. Nguyen, Natasha Alechina, and Brian Logan. Axiom pinpointing using an assumption-based truth maintenance system. In *DL*, 2012.
25. Bijan Parsia, Evren Sirin, and Aditya Kalyanpur. Debugging OWL ontologies. In *WWW*, pages 633–640, 2005.
26. Alessandro Previti and Joao Marques-Silva. Partial MUS enumeration. In *AAAI*, 2013.
27. Alan L. Rector and Ian R. Horrocks. Experience building a large, re-usable medical ontology using a description logic with transitivity and concept inclusions. In *Workshop on Ontological Engineering*, pages 414–418, 1997.
28. Raymond Reiter. A theory of diagnosis from first principles. *Artif. Intell.*, 32(1):57–95, 1987.
29. Stefan Schlobach and Ronald Cornet. Non-standard reasoning services for the debugging of description logic terminologies. In *IJCAI*, pages 355–362, 2003.
30. Stefan Schlobach, Zhisheng Huang, Ronald Cornet, and Frank van Harmelen. Debugging incoherent terminologies. *J. Autom. Reasoning*, 39(3):317–349, 2007.
31. Roberto Sebastiani and Michele Vescovi. Axiom pinpointing in lightweight description logics via Horn-SAT encoding and conflict analysis. In *CADE*, pages 84–99, 2009.
32. Roberto Sebastiani and Michele Vescovi. Debugging $\mathcal{EL}^+$ ontologies via SAT and SMT techniques. *Journal of Artificial Intelligence Research (submitted)*, 2014. Available from http://disi.unitn.it/~rseba/elsat/elsat_jair14_submitted.pdf.
33. Nicholas Sioutos, Sherri de Coronado, Margaret W. Haber, Frank W. Hartel, Wen-Ling Shaiu, and Lawrence W. Wright. NCI thesaurus: A semantic model integrating cancer-related clinical and molecular information. *Journal of Biomedical Informatics*, 40(1):30–43, 2007.
34. Evren Sirin, Bijan Parsia, Bernardo Cuenca Grau, Aditya Kalyanpur, and Yarden Katz. Pellet: A practical OWL-DL reasoner. *J. Web Sem.*, 5(2):51–53, 2007.
35. John Slaney. Set-theoretic duality: A fundamental feature of combinatorial optimisation. In *ECAI*, pages 843–848, 2014.
36. Kent A. Spackman, Keith E. Campbell, and Roger A. Côté. SNOMED RT: a reference terminology for health care. In *AMIA*, 1997.
37. Boontawee Suntisrivaraporn. *Polynomial-Time Reasoning Support for Design and Maintenance of Large-Scale Biomedical Ontologies*. PhD thesis, TU Dresden, Germany, 2009.